Forward Propagation Mathematics

1. Linear Transformation at Hidden Layer

The first step is a linear transformation of the input data X using the weights W_1 and biases b_1 of the hidden layer.

$$Z_1 = XW_1 + b_1$$

- X is the input matrix, where each row is an observation and each column is a feature.
- W_1 is the weight matrix of the first layer.
- b_1 is the bias vector of the first layer.
- Z_1 is the result of the linear transformation.

2. Activation Function at Hidden Layer

The next step is to apply the ReLU (Rectified Linear Unit) activation function to each element of Z_1.

$$A_1 = \mathrm{ReLU}(Z_1)$$
$$\mathrm{ReLU}(x) = \max(0, x)$$

- A_1 is the output of the hidden layer after applying ReLU.

3. Linear Transformation at Output Layer

Another linear transformation is applied to A_1 using the weights W_2 and biases b_2 of the output layer.

$$Z_2 = A_1W_2 + b_2$$

- W_2 is the weight matrix of the second (output) layer.
- b_2 is the bias vector of the second layer.
- Z_2 is the result of this linear transformation.

4. Softmax Activation at Output Layer

Finally, the softmax function is applied to Z_2. This is particularly used for multi-class classification to get the probability distribution over classes.

$$A_2 = \text{Softmax}(Z_2)$$
$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

- A_2 is the output of the network, representing the predicted probabilities for each class.
- The softmax function is applied to each row of Z_2, ensuring that the sum of probabilities in each row equals 1.

Summary of Forward Propagation

In summary, the forward pass involves these steps: input data is linearly transformed, an activation function is applied, another linear transformation occurs, and finally, the softmax function converts these outputs into probabilities. These operations collectively represent the forward pass of your neural network, transforming input data into predicted outputs.